

# **Checkout and Launch Control System (CLCS) System Test Plan**

**March 27, 1997**

Prepared By:

CLCS System Test Team  
CLCS Technical Integration  
DE-CLC  
Kennedy Space Center, FL 32899

# Checkout and Launch Control System (CLCS)

## System Test Plan

Prepared By:

*Original signed by*

---

David Reinhardt,  
System Test Engineer

*Original signed by*

---

Timothy McKelvey,  
System Test Engineer

Approved By:

*Original signed by*

---

Arnold Postell,  
Technical Integration Lead

*Original signed by*

---

Kirk Lougheed,  
System Engineering and Technical  
Integration Lead

*Original signed by*

---

Retha Hart,  
CLCS Program Manager

*Original signed by*

---

Tom Swanson,  
NASA Safety & Mission Assurance

Concurrence:

*Original signed by*

---

Ralph Esposito,  
USA CLCS Project Manager

*Original signed by*

---

Mark Dotterweich,  
LMSMS CLCS Project Manager

*Original signed by*

---

Jeff Wheeler,  
NASA User Liaison

*Original signed by*

---

Chris Best,  
USA User Liaison

## Table of Contents

<b>1.0 SCOPE .....</b>	<b>1-1</b>
1.1 IDENTIFICATION .....	1-1
1.2 PURPOSE.....	1-1
1.3 SYSTEM OVERVIEW .....	1-1
1.4 DOCUMENT ORGANIZATION.....	1-1
<b>2.0 APPLICABLE DOCUMENTATION.....</b>	<b>2-1</b>
2.1 PARENT DOCUMENTS.....	2-1
2.2 APPLICABLE DOCUMENTS.....	2-1
2.3 REFERENCE DOCUMENTS.....	2-1
<b>3.0 TEST PLAN DESCRIPTION.....</b>	<b>3-1</b>
3.1 CLCS TEST PHILOSOPHY .....	3-1
3.1.1 CLCS Test Approach Overview.....	3-1
3.1.2 CLCS Test Definitions .....	3-2
3.1.3 CLCS System Validation and Acceptance Testing .....	3-5
3.2 CLCS TESTING PROCEDURES AND GROUND RULES .....	3-7
3.2.1 Test Flow .....	3-7
3.2.1 System Test Reviews and Documents.....	3-7
3.2.2 Roles and Responsibilities .....	3-9
3.2.3 Test Exception Procedures .....	3-11
3.2.4 Test Change Procedures .....	3-11
3.2.5 Problem Reporting .....	3-11
<b>APPENDIX A - ACRONYMS AND DEFINITIONS .....</b>	<b>A-1</b>

**Table of Figures**

Figure 1. CLCS Development Test Approach ..... 3-1

Figure 2. Test and Certification Philosophy ..... 3-5

Figure 3. Test Development Flow..... 3-7

Figure 4. CLCS Problem Reporting..... 3-12

## **1.0 Scope**

This document defines the system test plan for use in the development and delivery of the Checkout and Launch Control System (CLCS).

### **1.1 Identification**

This document is the Checkout and Launch Control System (CLCS) System Test Plan Document, 84K00056 Rev. BASIC. This document will be maintained under the control of the CLCS configuration management program.

### **1.2 Purpose**

The purpose of this document is to define the plan to be used by CLCS Systems Integration and Test engineers to ensure that the CLCS is thoroughly tested at the system level. As the development of the CLCS progresses, this document will be updated as required.

### **1.3 System Overview**

The CLCS is composed of a Real Time Processing System (RTPS) and a support system called the Shuttle Data Center (SDC). The RTPS provides the capability to checkout and control the elements of the current Space Transportation System (STS) and Ground Support Equipment (GSE). It provides support to the Space Shuttle Program into the 21<sup>st</sup> Century and a basic infrastructure upon which to base future design projects such as the Orbiter Upgrades and RLV.

The CLCS replaces the current Launch Processing System (LPS) with state-of-the-art Commercial Off the Shelf (COTS) based technology. Wherever possible, COTS software is used instead of developing software where the requirements of the task are satisfied by the COTS software products. Any developed software will be written in high level languages that have demonstrated a high degree of portability between platforms. COTS hardware is also utilized as much as possible in the CLCS. This strategy provides a highly reliable system that is both supportable, with minimal software and hardware upgrade impacts, and expandable, with a solid base design.

### **1.4 Document Organization**

This document is divided into three sections and one appendix:

Section 1, Scope, discusses the purpose of the CLCS System Integration Operations Test, provides a system overview, and describes software and hardware configurations for the system.

Section 2, Applicable Documentation, lists the documents used to create and support this document.

Section 3, Test Plan Description, contains a detailed description of the CLCS approach to system level testing.

Appendix A, Acronyms and Definitions, contains a listing of acronyms and selected word definitions (for words that may have multiple interpretations).

## 2.0 Applicable Documentation

The following documents, of the latest revision, form a part of this document to the extent specified.

### 2.1 Parent Documents

The documents in this paragraph establish the criteria and technical basis for the existence of this document. The parent documents are:

Parent Document	Document Number
CLCS Program Management Plan	<TBS>
CLCS System Engineering Management Plan (SEMP)	<TBS>
CLCS Project Plan	<TBS>

### 2.2 Applicable Documents

Applicable documents are those documents which form a part of this document. These documents, at the revisions listed below, carry the same weight as if they were stated within the body of this document.

Applicable Document	Document Number
CLCS Certification Plan	<TBS>
CLCS CM Plan	<TBS>
CLCS Integration Management Plan	<TBS>

### 2.3 Reference Documents

Reference documents are those documents which, though not a part of this document, serve to clarify the intent and contents of this document.

Reference Document	Document Number
CLCS System Level Specification	<TBS>
CLCS System Design Document	<TBS>
CLCS Concept of Operation	<TBS>

## 3.0 Test Plan Description

### 3.1 CLCS Test Philosophy

#### 3.1.1 CLCS Test Approach Overview

This section provides an overview of all testing in the development of the CLCS. While this test plan covers only system testing (referred to as System Test or ST), it is important to understand the ST in the context of the overall development test flow.

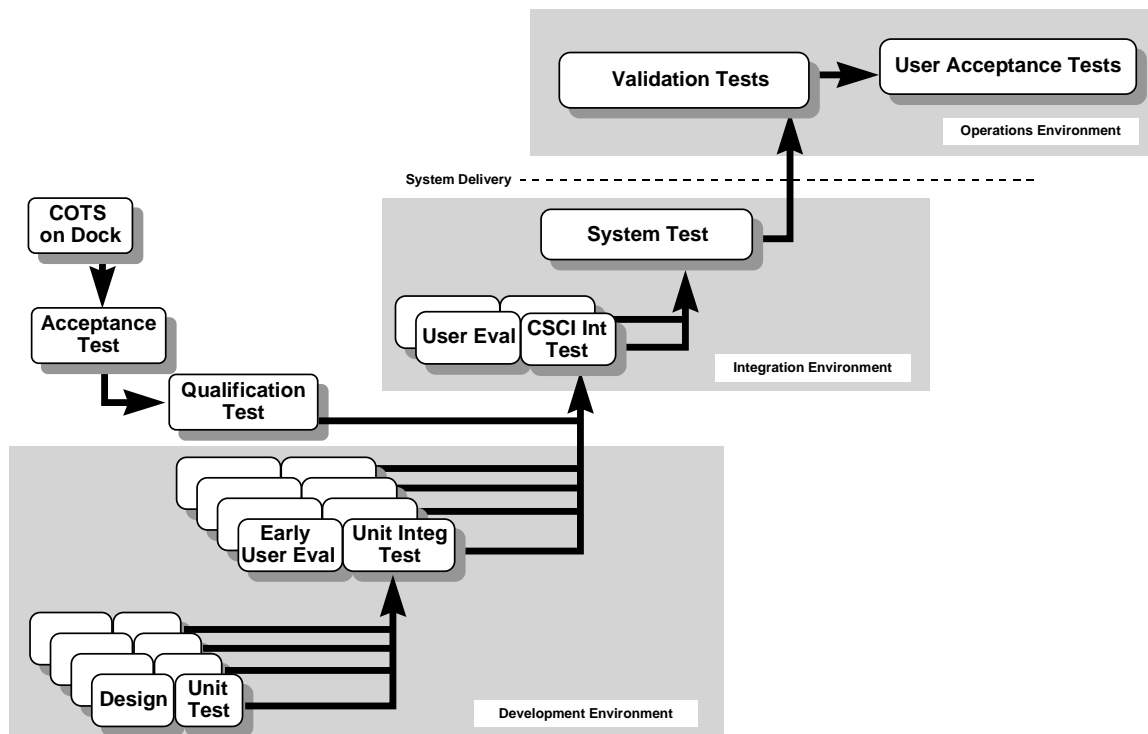


Figure 1. CLCS Development Test Approach (One Delivery Cycle Shown)

CLCS development testing starts at the lowest level (Unit Tests and COTS Acceptance Tests) and goes through System Testing. Validation and User Acceptance Testing are performed outside of the development organization and are key parts of the system certification process. Basic guidelines for all levels of development testing include the following:

- Each test is developed and reviewed in relation to previous testing. Unit Integration Tests (UIT's), CSCI Integration Tests (CIT's), and System Test's do not simply repeat procedures performed in previous tests.



- Test procedures or procedure segments from earlier deliveries (at the same test level) are modified as necessary and reused where possible, e.g., tests from a given System Test may be used in a subsequent System Test.
- Early user evaluation will be an integral part of the test development process:
  - Feed back from user experience with early versions of software will allow for the development of more focused tests at the UIT level and above
  - Areas of concern to the user community will be emphasized
  - Early evaluation will facilitate coordination of the content of different levels of testing
- End-to-end testing process will be consistent across all software types:
  - System Software
  - System Application Software
  - User Application Software (e.g., Re-engineered “GOAL type” applications)
- End-to-end testing process will be consistent across all development organizations
- Test procedures will be carefully reviewed to ensure that personnel, hardware and software are not jeopardized during testing.
- There will be fully documented and approved test procedures for each level of testing. Test documents will be controlled under the CLCS project CM process.

### 3.1.2 CLCS Test Definitions

#### 3.1.2.1 Unit Test

Unit Tests (UT's) are performed by the software developer to verify basic functionality of a demonstrable segment of a Computer Software Configuration Item (CSCI); for example, a single program that can be meaningfully executed in a “stand alone” mode. Unit tests require a minimal number of stubbed interfaces and are tested against Functional Requirements (as documented in the Hardware Requirements and Design Specifications and/or the Software Requirements and Design Specifications). This is the first level of testing for new software, and may be omitted for more mature software (e.g., reused code, incremental updates to code that has previously been Unit Tested and does not require regression testing). If the UT is omitted, the code proceeds to the next level of test, the Unit Integration Test. Omission of UT will be determined by the development lead and reviewed by S&MA on a case by case basis. The level of formality (amount

and type of documentation) is minimal. The tests are signed off by the development lead. S&MA may review the test procedures and may or may not witness/signoff the tests. Unit tests are performed in the Development Environment.

#### 3.1.2.2 Unit Integration Test

Unit Integration Tests (UIT's) are performed by the software developer to verify basic functionality and successful integration of a set of programs (Computer Software Component, CSC) with its new and/or modified units. The UIT's are performed against Functional Requirements and can be the first level of testing for reuse software. UIT's are witnessed and signed-off by development lead. UIT's are witnessed and signed-off by S&MA when the test is formally demonstrating compliance with Functional Requirements. UIT's are performed in the Development Environment.

#### 3.1.2.3 Acceptance Test (COTS)

Acceptance Tests (AT's) are performed by the procuring organization to verify that COTS hardware and/or software meet the requirements of the procurement specification. This is the first level of testing for COTS hardware and software, and is roughly equivalent to a Unit Integration Test.

#### 3.1.2.4 Qualification Test (COTS)

Qualification Tests (QT's) are performed by the procuring organization to verify that COTS hardware and/or software meet the CLCS platform requirements. This is the final level of standalone testing for COTS hardware and software, and is roughly equivalent to a CSCI Integration Test.

#### 3.1.2.5 CSCI Integration Test

CSCI Integration Test's (CIT's) are performed by the software developer (with support from the System Integration group) to verify basic functionality and successful integration of a functionally complete set of programs (e.g., CSCI's, such as Application Software, Application Services, System Services) with its new and/or modified units in an operational like environment. CIT's are the final tests against Functional Requirements and are the final level of testing prior to System Tests. The tests are reviewed by the development lead, S&MA, System Integration and System Test, and are witnessed and signed-off by the development lead and S&MA. CIT's are performed in the Integrated Development Environment (IDE), which is configured the same as the operational environment and is used as a dedicated system test facility.

### 3.1.2.6 System Test

The System Tests (ST's) are performed by the System Test team (which is a subset of the System Integration and Test group) to verify successful integration of a system delivery and to demonstrate that the system is "operable" in all required modes (e.g., development, maintenance, system operations, test/launch support, etc.). The System Test demonstrates that the development organization has met the agreements made with the user community (in the form of requirements) regarding the capabilities in a delivery. The ST's validate system level requirements and external interfaces emphasizing system functionality, performance and system wide data flow. They typically will include Regression Testing for previously delivered capabilities (Regression Testing will occur at the System Test for every delivery). The ST's are performed against System Level Specification Requirements and any functional level requirements that have not been covered by CIT's. They are reviewed by S&MA, System Integration, development leads and they are witnessed and signed-off by System Test and S&MA. The nature of the incremental CLCS development process will lead to deliveries that only partially address a given system level requirement. The system test team will maintain records of partially demonstrated system level requirements and ensure that the requirements will be fully met by the end of the project. The number of tests and test cases for ST's are finalized after the last Detailed Design Panel for a given delivery and are based on threads, operational scenarios and operations constraints. This is the final level of testing prior to putting the system in an operational like environment for user validation and certification.

### 3.1.3 CLCS System Validation and Acceptance Testing

The CLCS end user community's portion of the certification process verifies that the development organization has delivered a functionally complete and correct system which can be used as safely and reliably as possible to support all operations. This process is under the control of the end user community.

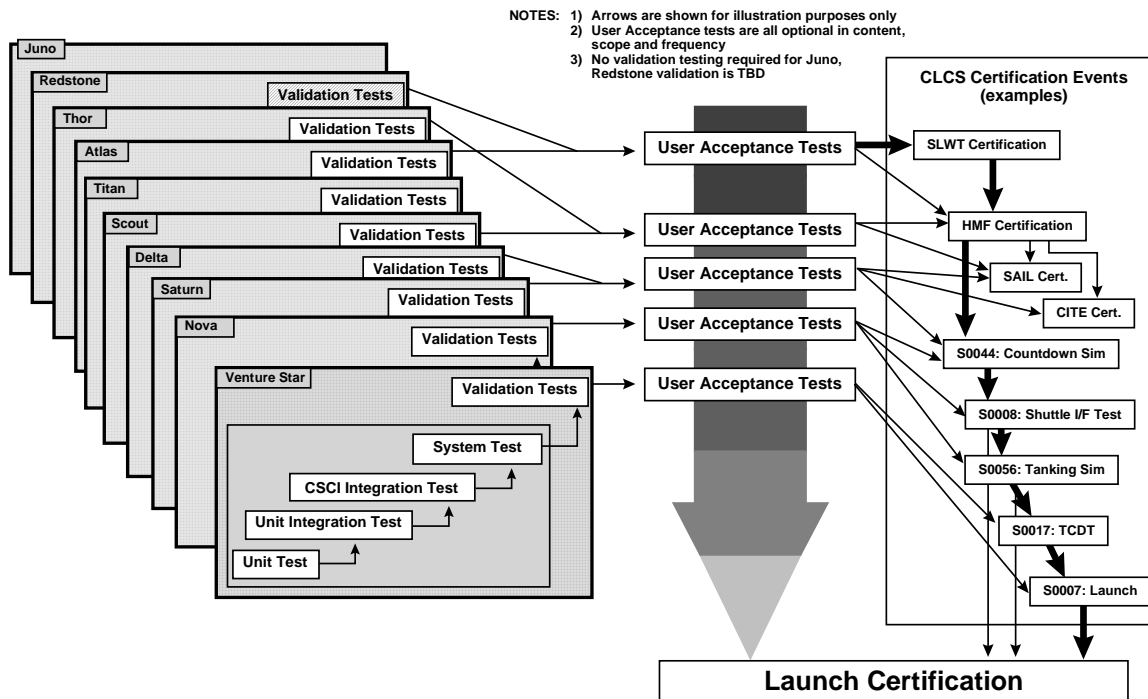


Figure 2. Test and Certification Philosophy

#### 3.1.3.1 Validation Tests

Validation tests are performed by an independent group at KSC and are intended to verify that the product delivered to the operations organization from the development organization meets the users' requirements. Validation tests are executed for each delivered system. They occur in the Operations Environment (prior to certification) as appropriate, determined by the testing organization. Validation tests are supported by development leads and System Integration and System Test.

#### 3.1.3.1 User Acceptance Tests

User Acceptance Tests (UAT's) verify that the system is processing data correctly (data and displays are valid) and that the system can be certified for use in approved types of operations (e.g., test/checkout, launch, landing). The scope of the user acceptance tests is defined by the user community on a per test basis.

User acceptance tests are executed in the operations environment as required by users (not necessarily at each delivery), usually in preparation for operational certification. User acceptance tests are supported by development leads and System Integration and System Test. This is the final level of test prior to full operations support.

The following table shows roles and responsibilities of types of organizations in regard to the types of testing:

Test	Development Organization	System Integration Group	System Test Group	End User Community
Unit Test	R			
Unit Integration test	R			
CSCI Integration Test	R	S		
CIT Functional Testing	R	S		
CIT Regression Testing	S	S	R	
System Test		S	R	
Validation Test			S	R
User Acceptance Test			S	R
COTS				
Acceptance Test	R			
Qualification Test	R	S		

R = Responsible, S = Support

## 3.2 CLCS Testing Procedures and Ground Rules

### 3.2.1 Test Flow

System tests for each CLCS delivery will, where practicable, conform to the following schedule template:

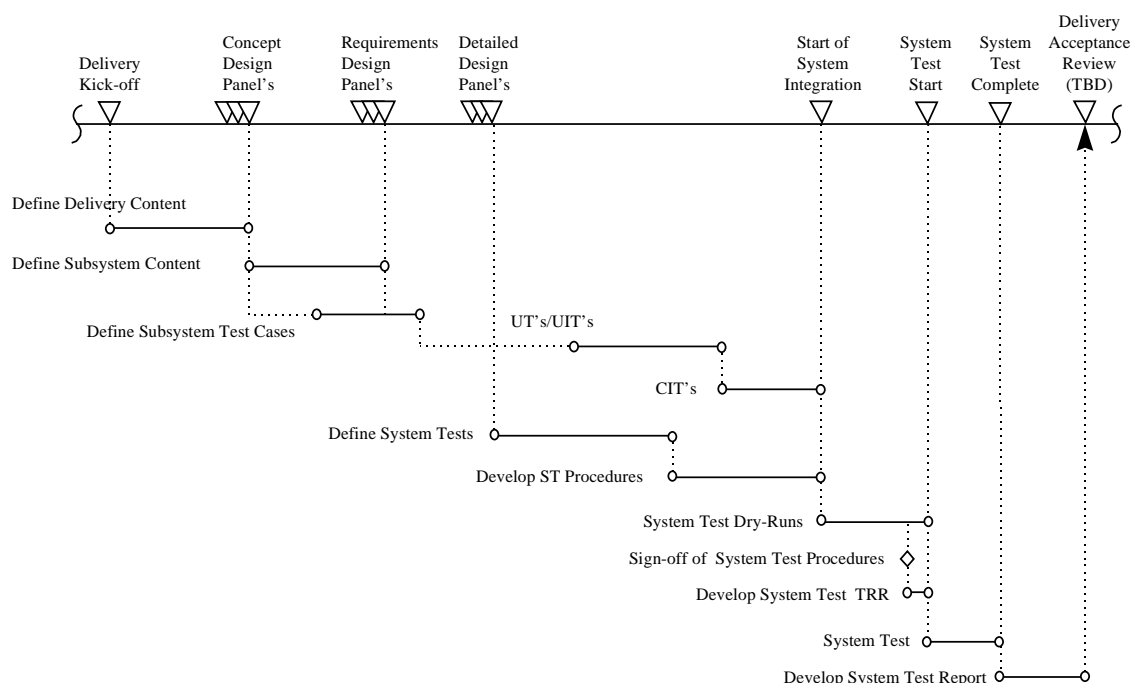


Figure 3. Test Development Flow

Development of System Test procedures begins after the last Detailed Design Panel for a given delivery, though the System test team will participate in the prior design reviews. At this time, the test cases will be defined and then approved by the Technical Integration Lead with concurrence of the System Engineering & Integration leads and the User Liaisons. Procedure development will be done incrementally starting with test case development and finishing with dry-runs prior to final approval of the procedures and execution of the test. Subsystem level testing will be taken into account, as will inputs from early user evaluations of the delivery.

### 3.2.1 System Test Reviews and Documents

#### 3.2.1.1 Test Readiness Review

The Test Readiness Review (TRR) is presented, if deemed necessary by the Test Conductor, prior to the start of a system test. The purpose of the presentation is to describe the state that the system is in and whether or not it is ready for testing.

The TRR is chaired by the NASA CLCS Technical Integration lead and is presented by the CLCS System Test team. The goal of the review is to obtain concurrence from the NASA CLCS Technical Integration lead and from S&MA that the system is ready for system level testing. This concurrence provides authority to commence testing. The TRR presentation includes the following information:

- Delivery content summary
- Target system configuration for the delivery
- System configuration to be used during the test
- Summary of previous testing (subsystem and system level tests)
- Summary of dry run testing
- Known problems
- Ground rules for test conduct
  - Roles and responsibilities
  - Handling test exceptions, including emergency procedures as necessary
  - Problem reporting procedures
- Test schedule and locations

#### 3.2.1.2 System Test Procedures

The System Test Procedures Documents will define:

- Test scope and intent
- Test cases and test dependencies
- Tracing of system level requirements to test cases
- Specific test procedures and sign off

#### 3.2.1.3 Daily Test Progress Review

If necessary, in the opinion of the Test Conductor or the S&MA witness, a daily Test Progress Review (TPR) presentation will be provided by the Test Conductor at the beginning of each day of testing. This review will cover test progress to date, summary of problems found during the previous day(s) of testing and tests planned for that day.

#### 3.2.1.4 Post-Test Review

At the completion of system testing, a Post-Test Review (PTR) will be held (at the option of the Test Conductor and/or the S&MA witness). This review will summarize the testing completed, problems found during testing and any follow up actions required to declare the successful completion of the test.

### 3.2.1.5 Test Report

The Test Report (TR) will be released after the completion of all System Testing and will be included in the delivery package for the given system delivery. This report will include a summary of the test procedures, the test results, problems found and a copy of the “as run” test procedures, including any additions or modifications to the procedures generated during the test.

This report is the System Test organization’s input to the Delivery Acceptance Review (DAR). The DAR is a meeting, chaired by the CLCS Delivery Manager, at which the delivery package is reviewed and the Delivery is declared complete. The delivery package includes such things as magnetic tape copies of software to be delivered, all development documentation (users guides, requirements, design documents), all test documentation (subsystem and system level tests) and known problems.

### 3.2.2 Roles and Responsibilities

The following matrix shows the roles and responsibilities for each organization involved in CLCS System Testing. Following that are the detailed descriptions of those responsibilities.

Organization	System Test Plan	System Test Procs.	ST Dry Run	TRR	ST Exec.	TPR	PTR	Test Report
CLCS Developers	R	R	S					
CLCS System Integration	R	R	S,E	S	S	S	S	R
CLCS System Test	G	G	E	G,E	E	G,E	G,E	G
CLCS System Admin		R	S	S	S	S	S	
NASA S&MA	R,A	R,A		R,A	W	R,A	R,A	R,A
USA S&MA	R	R		R		R	R	R
LMSMS QA	R	R						R
End User Community	R	R		S	S	S	R	R
Operations Support			S	S	S	S	S	

W = witness (with signature approval), E = execute, R = review/attend (comments), A = approve (signature), S = support, G = generate

#### 3.2.2.1 CLCS Development Organization Responsibilities

CLCS Developers (programmers, CSCI leads) - Review and support development of system test procedures. This includes the development of test data, scripts and procedures, as well as providing documentation of previous subsystem level testing and which functional requirements have been met. Support dry runs and support test execution as necessary.



CLCS Systems Integration (system integrators, Technical Integration Lead) - Review and support development of system test procedures, ensure that any required pre-test system builds have been completed, support dry runs and support test execution as necessary (e.g., problem isolation during system test execution).

CLCS System Test (System Test Test Conductor, testers, Technical Integration Lead) - Generate all system test documents, execute the system tests and perform all test reviews. Act as Test Conductor. The Test Conductor is in charge of the execution of a System Test. The Test Conductor is responsible for ensuring that the system is in the proper configuration for the test, that all required approvals have been obtained and that the test support personnel and tools are in place. The Test Conductor is the final authority as to the conduct of the test, including changes to the test procedures, documenting problems found and changes in the system configuration during the test.

CLCS System Administration - Certain test procedures will require that the tester have the ability and permission to use administrative tools and procedures (“root” access). The system administrators will support the development and execution of these tests.

#### 3.2.2.2 Safety and Mission Assurance Organization Responsibilities

NASA S&MA - NASA S&MA representatives will approve test plans and procedures, witness and approve tests, approve any changes to test procedures and configurations and have the authority to stop testing if significant problems are found.

USA S&MA - Support NASA S&MA as required.

LMSMS QA - Support NASA S&MA as required.

#### 3.2.2.3 End User Community Responsibilities

The end user community (primarily System Engineers) is expected to review test cases and support their development. They also may be requested to support test procedure development to ensure conformance with operational concepts, scenarios and plans. End users will also review test results.

#### 3.2.2.4 Other Support Personnel Responsibilities

Operations Support - Certain test procedures will require the use of operational facilities and capabilities such as hardware configuration/reconfiguration and accessing live or simulated data. The operations support organization will support the development and execution of these test procedures as necessary to support a given test configuration.

#### 3.2.3 Test Exception Procedures

In the event of test exceptions (i.e., test results that do not match the expected results), the Test Conductor has the responsibility to see that the appropriate problem reports (PR's) are generated, properly classified regarding impact on both the delivery and on the continuation of the test, and properly classified regarding urgency. The Test Conductor will, if warranted in his/her opinion, suspend testing and schedule the resumption of the test as soon as possible. Otherwise, the PR will be generated, classified and submitted, testing will resume and the PR will be reported as required in the TPR, PTR, TR and as otherwise required by the CLCS Safety and Mission Assurance Plan.

#### 3.2.4 Test Change Procedures

The Test Conductor has the sole authority to add to, delete from or red-line the test procedures during the execution of the test (though the changes may be developed by anyone supporting the test). All changes must be approved by the S&MA representative. If, in the opinion of the Test Conductor or the S&MA representative, the changes are excessive or have the potential to endanger personnel, hardware or software, a problem report can be written against the test procedures. In this case, the test would proceed as written as much as practicable and the updated test procedures would be executed in a separate test which in turn would close the PR.

#### 3.2.5 Problem Reporting

During development, informal problem reports will be generated and tracked by the developers. These informal PR's will not be reported except as deemed necessary by the developers. When software is ready to start CIT, existing informal PR's will be converted to formal PR's and any new problems will be documented with formal PR's. Formal PR's will be reported to CLCS project management on a regular basis and will be included in all delivery documentation. At the end of all CLCS development, all existing formal PR's will be converted to the PRACA system and the CLCS PR system will be discontinued.

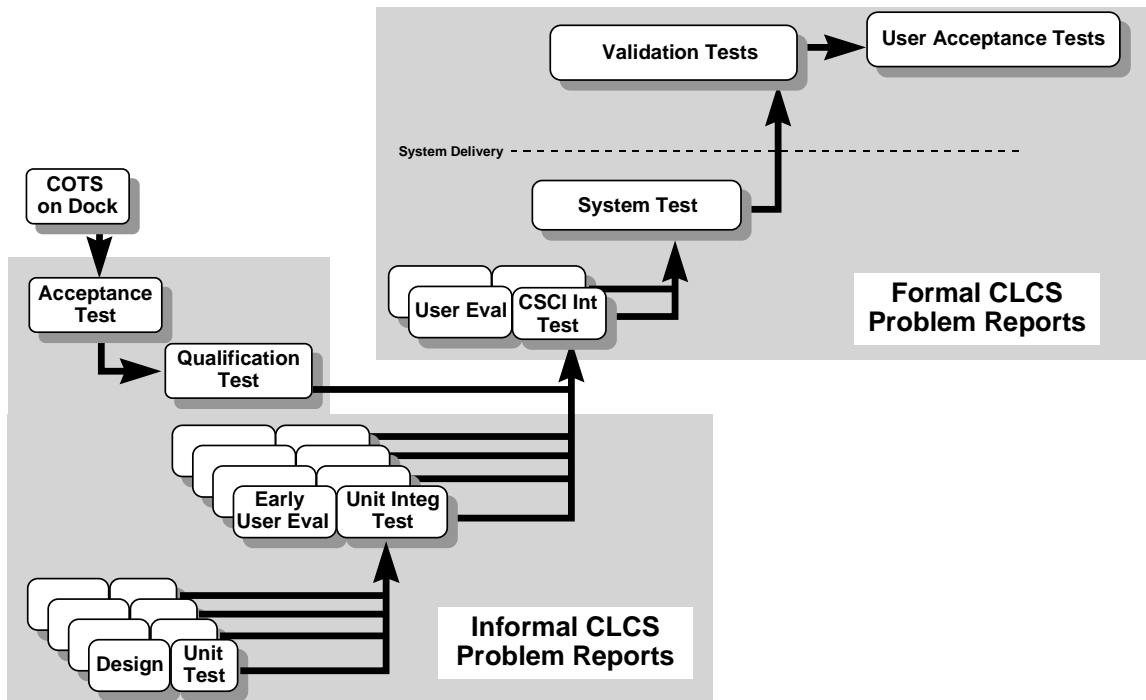


Figure 4. CLCS Problem Reporting

The following definitions will be used for problem reports found prior to and during System Testing. Note that these may not be the same definitions as those used in an operational environment:

- **Critical:** A critical problem is one in which a user (or the system on behalf of a user) is unable to perform a required task, and no viable workaround exists. Problems involving safety are considered to be critical.
- **Major:** A major problem is one in which a user is unable to perform a required task, but a workaround is available, even though the workaround may impact user functions/operations.
- **Minor:** A minor problem is one with minimal impact to the users, a workaround is available or not required and does not significantly affect the ability of the users to perform their function.

The above PR classification refers only to functionality. Urgency in obtaining a correction to the problem is not considered in this classification.

## Appendix A - Acronyms and Definitions

AT	Acceptance Test - Test to accept hardware and software from a vendor
Certification	Final approval to use a system for a specified set of operations (e.g., hazardous operations in the HMF, launch operations, etc.)
CI	Configuration Item
CIT	CSCI Integration Test
CLCS	Checkout and Launch Control System
CM	Configuration Management
COTS	Commercial Off The Shelf
CSC	Computer Software Component
CSCI	Computer Software Configuration Item
DAR	Delivery Acceptance Review
EDL	Engineering Development Laboratory
GSE	Ground Support Equipment
HCI	Human Computer Interface
HMF	Hypergol Maintenance Facility
HW	Hardware
HWCI	Hardware Configuration Item
IDE	Integrated Development Environment
I/F	Interface
KSC	Kennedy Space Center
LAN	Local Area Network
LCC	Launch Control Complex
LMSMS	Lockheed Martin Space Mission Systems and Services
LPS	Launch Processing System
NASA	National Aeronautics and Space Administration
MSC	Mission Systems Contract (held by LMSMS)
OS	Operating System
PTR	Post-Test Review
PR	Problem Report

QA	Quality Assurance
QE	Quality Engineering
QT	Qualification Test
RLV	Reusable Launch Vehicle
RTPS	Real Time Processing System
RVM	Requirements Verification Matrix
SDC	Shuttle Data Center
SDE	Satellite Development Environment
SEMP	System Engineering Management Plan
SFOC	Space Flight Operations Contract (held by USA)
ST	System Test
SLWT	Super Light Weight Tank
S&MA	Safety and Mission Assurance (includes Reliability, Maintainability, Safety and Quality Assurance)
STS	Space Transportation System
SW	Software
TC	Test Conductor
TPR	Test Progress Review
TR	Test Report
TRR	Test Readiness Review
UAT	User Acceptance Test - Test performed by user community post delivery as part of the system certification process
UIT	Unit Integration Test
USA	United Space Alliance
UT	Unit Test
Validation	Testing performed by organization(s) outside of the developing organization to ensure that the delivered system processes data correctly and conforms to the operations concepts